

## University of Groningen

### **Cognitive control explains the mutual transfer between dimensional change card sorting and first-order false belief understanding:**

Arslan, Burcu; Verbrugge, Rineke; Taatgen, Niels

*Published in:*  
Biologically Inspired Cognitive Architectures

*DOI:*  
[10.1016/j.bica.2017.03.001](https://doi.org/10.1016/j.bica.2017.03.001)

**IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.**

*Document Version*  
Final author's version (accepted by publisher, after peer review)

*Publication date:*  
2017

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*

Arslan, B., Verbrugge, R., & Taatgen, N. (2017). Cognitive control explains the mutual transfer between dimensional change card sorting and first-order false belief understanding: A computational modeling study on transfer of skills. *Biologically Inspired Cognitive Architectures*, 20, 10-20.  
<https://doi.org/10.1016/j.bica.2017.03.001>

#### **Copyright**

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

#### **Take-down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

**Cognitive Control Explains the Mutual Transfer Between  
Dimensional Change Card Sorting and First-order False Belief Understanding:  
A Computational Modeling Study on Transfer of Skills**

Burcu Arslan<sup>1\*</sup>, Rineke Verbrugge<sup>1</sup>, Niels Taatgen<sup>1</sup>

Institute of Artificial Intelligence, University of Groningen, P.O. Box 407,  
9700 AK Groningen, The Netherlands

\*Corresponding author

e-mail: [b.arslan@rug.nl](mailto:b.arslan@rug.nl)

## **Abstract**

While most 3-year-olds fail both in the false belief task of theory of mind and Dimensional Change Card Sorting task of cognitive control, most 4-year-olds are able to pass these tasks. Different theories have been constructed to explain this co-development. To investigate the direction of the developmental relationship between false belief reasoning and cognitive control, Kloo and Perner (2003) trained 3-year-olds on the false belief task in one condition and on the Dimensional Change Card Sorting task in another condition. They found that there is a mutual transfer between the two tasks, meaning that training children with the Dimensional Change Card Sorting task with feedback significantly improved children's performance on the false belief task and vice versa. In this study, we aim to provide an explanation for the underlying mechanisms of this mutual transfer by constructing computational cognitive models. In contrast to the previous theories, our models show that the common element in the two tasks is two competing strategies, only one of which leads to a correct answer. Providing children with explicit feedback trains them to use a strategy of control instead of using a simpler reactive strategy. Therefore, we propose that children start to pass the false belief and cognitive control tasks once they learn to be flexible in their behavior depending on the current goal.

**Keywords:** false belief reasoning; cognitive control; transfer of skills; computational cognitive modeling; cognitive development; PRIMs

## Introduction

There are many hilarious videos on the Internet showing 2- and 3-year-olds' failure on the *hide and seek* game and on the *marshmallow* test. On the other hand, most 4-year-olds are able to hide themselves at a place where the seeker cannot find them immediately in the *hide and seek* game. In the *marshmallow* test, most 4-year-olds are able to wait for the experimenter to come back to the room in order to get more marshmallows instead of eating one marshmallow right away. The key element of success in the *hide and seek* game is to be able to take the perspective of the seeker and the key element in the *marshmallow* test is to have self-control.

In line with these videos, a number of correlational studies have shown that there is a relation between children's development of *theory of mind* and *cognitive control* (Perner & Lang, 1999; Müller, Zelazo, Imrisek, 2005; Henning, Spinath, & Aschersleben, 2011). Theory of mind can be defined as a general term for perspective taking by reasoning about others' representational mental states such as beliefs, desires and knowledge (Premack & Woodruff, 1978). Cognitive control, which is an important component of executive functions, can be defined as the ability to flexibly select actions in the furtherance of chosen goals, instead of inflexibly reacting to the environment while ignoring the current goal. Therefore, cognitive control requires selecting appropriate information related to the current goal for processing and inhibiting inappropriate information and responses. For example, to succeed in the *marshmallow* test, children have to inhibit the urge to eat the marshmallows right away and have to consider the current goal, which is waiting for the experimenter in order to receive a larger award. Similarly, if an agent's initial goal is to find another agent who has blue eyes and if the current goal is finding an agent who has brown shoes, then the agent should ignore the eye color of other agents and attend to the agents' shoe color.



There are three main theories about the relation between theory of mind and cognitive control<sup>1</sup>. The *Cognitive Complexity and Control-revised* theory (CCC-r; Zelazo, Müller, Frye, & Marcovitch, 2003) suggests that the common component between theory of mind and cognitive control is representational and also related to the activation and inhibition of rules. According to this theory, theory of mind and cognitive control tasks develop together because they both require a child to reason by using embedded *if-if* rules and both need inhibition of rules. The second theory suggests that being able to take the perspective of others improves children's cognitive control abilities, meaning that there is transfer of skills from theory of mind to cognitive control (Perner, 1998). On the contrary, the third theory suggests that the direction of transfer is from cognitive control to theory of mind (Russell, 1996 as cited in Kloo & Perner, 2003).

Although correlational studies have shown that children's theory of mind and cognitive control abilities co-develop, as reflected in the second and third theories, there is no consensus on the direction of this relationship. In order to investigate the direction of the relationship, Kloo and Perner (2003) conducted a training study with children by using a theory of mind task and a cognitive control task. We provide the details of these tasks in the following subsection. Kloo and Perner's results showed that there is a mutual transfer between cognitive control and theory of mind, meaning that training children with a cognitive control task with feedback significantly improved children's performance on a theory of mind task and vice versa.

Based on these findings, Kloo and Perner propose that the common component between the two tasks is representational. Differently from CCC-r theory, they argue that the problem 3-year-olds encounter is related to failure in redescribing an object or situation and that training children with explicit feedback helps them to understand that an object or certain situation can be described differently from different perspectives. However, Kloo and Perner stated that the exact nature of transfer effect remains to be determined.

---

<sup>1</sup> see Carlson, Moses, & Hix, 1998; Leslie & Polizzi, 1998; Carlson, Moses, & Breton, 2002 for other theories that are related to the role of other components of executive functions, such as inhibition and working memory.

The main goal of the current study is to provide an explanation for the nature of the mutual transfer between cognitive control and theory of mind by constructing computational cognitive models.

How does training children help transfer of skills? According to the *primitive information processing elements* theory (PRIMs; Taatgen, 2013), there are two explanations for the transfer of skills that can be modeled with the same mechanism. According to Explanation 1, skills can transfer from one task to another when those tasks have a substantial overlap in their procedural knowledge. For example, multi-column multiplication shares knowledge with multi-column addition, and many other pen-and-paper arithmetic algorithms. Acquiring this knowledge is a relatively slow process. On the other hand, Explanation 2 assumes that the knowledge for both tasks is already present in memory: it just has to be mobilized at the right moment. Suppose a particular task has two possible strategies, A and B, and suppose B is superior to A, but A is simpler. If parts of strategy B, in particular the parts that are necessary to select B, are trained in another task, it becomes more likely that strategy B will be chosen over strategy A. Our models are based on Explanation 2, because the training time in the experiment is relatively short.

In the following subsection, we first present the details of the theory of mind and cognitive control tasks that were used in Kloo and Perner's training study together with a summary of the design of the study, in order to provide a sufficient background to understand our computational cognitive models and to interpret the simulation results.

### **Kloo and Perner's training study**

Kloo and Perner's training study (Experiment 2) tested a sample of 44 children between the ages three and four ( $M = 45.1$  months,  $SD = 4.9$  months) at four different sessions almost one week apart from each other: i) pre-test, ii) training day 1, iii) training day 2, and iv) post-test.

At the pre-test and post-test sessions, children were tested with a standard theory of mind task and a cognitive control task together with a verbal intelligence task.

As a theory of mind task, Kloo and Perner used a standard false belief task (FB; Wimmer & Perner, 1983), which is one of the most commonly used tasks to assess young children's development of theory of mind. During the FB task, children listened to a story accompanied by illustrations showing that a protagonist placed an object into a location, after which that object was moved to another location while the protagonist was not present. Children had to *predict* where the protagonist would look for the object based on the protagonist's false belief, instead of reporting their own true belief about the location of the object. After that, children were shown another picture of the protagonist searching for the object based on her false belief (empty location) and were asked to *explain* the behavior of the protagonist. The same type of story with a different object and protagonists was used at the post-test session. On each false belief task, children's scores were between 0 and 2 based on their answers for the prediction and explanation questions, not their explanations. Children did not get any feedback at the pre-test and post-test sessions. Note that Kloo and Perner also reported children's performance on the predictions separately. For the purpose of our study, we only modeled children's predictions.

As a cognitive control task, they used the Dimensional Change Card Sorting task (DCCS; Frye, Zelazo, & Palfai, 1995). In the standard version of the DCCS task, children are presented with two target cards, one on the left and the other one on the right. After that, an experimenter introduces a set of test cards. The test cards have two dimensions, one of which matches with one target card and the other matches with the other target card (see Figure 1). At the beginning of the experiment (pre-switch phase), children are introduced to the rule of the "Animal" game. In the "Animal" game, children are expected to sort the test cards by pointing to the target that matches to the test card with the animal type. For example, if the test card "small horse" is shown, the children are expected to point to the target card "big horse", which is on the left. After playing six trials of the "Animal" game, the experimenter introduces the new "How-Big" game (post-switch phase). At the post-switch phase, children have to sort the test cards based on the other dimension, namely size, also for

again six trials by pointing to the target card that matches to the test card with size. For example, if the test card “small horse” is shown, children are expected to point to the “small fish” target card, which is on the right. Even though most children around the age of three do not have major problems in sorting the cards correctly at the pre-switch phase, after the rule changes, they keep sorting the cards by the pre-switch rule instead of the new post-switch rule. On the other hand, similar to 4-year-olds’ development of false belief reasoning, most children around the age four are able to sort the cards correctly at the post-switch phase as well (Doebel & Zelazo, 2015).

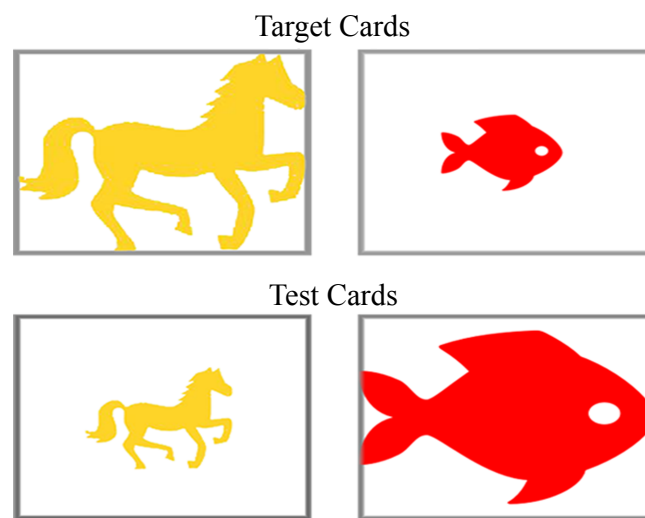


Figure 1. An example of the DCCS task. In this example, if the game is an “Animal” game, children are expected to sort the test card “small horse” by pointing to the target card “big horse”, which is on the left and to sort the test card “big fish” by pointing to the target card “small fish”, which is on the right. If the game is a “How-Big” game, children are expected to sort the test card “small horse” by pointing to the target card “small fish”, which is on the right and to sort the test card “big fish” by pointing to the target card “big horse”, which is on the left.

Different from the pre-test, children were tested with a three-boxes version of the standard DCCS task at the post-test session, which had three target cards instead of two. The reason for using three target cards at the post-test session was to control for children’s usage of a reversal shift strategy, which is pointing out the opposite target card. Children were expected to sort six cards at the post-switch phase, both in the standard DCCS and three-boxes version of the DCCS tasks. The experimenter

sorted the first card as an example; therefore, children's score was between 0 and 5.

At the training sessions, children were assigned to one of the following three training groups: i) DCCS (N = 14), ii) FB (N = 15), iii) control (N = 15). Children in the DCCS group were trained with a DCCS task with three dimension switches (i.e., color, number, color, number). Subsequently, children were introduced with a new set of test cards while the target cards were the same and they were expected to sort the new test cards again first by color, then by number. Finally, a new set of test cards was introduced with new target cards and children were again expected to sort the cards first by color and then by number. Therefore, DCCS training consisted of ten switches in total for both training days. The experimenter provided positive and negative feedback by emphasizing which game they were playing and how they should sort the cards at each ten switches.

The crucial parts of the feedback for our model of DCCS at the training session are the parts in which children were reminded that they were not playing the pre-switch phase game anymore and were asked questions about which game they were playing at the post-switch phase (e.g., "... However, we are not playing the "Animal" game, the game with "horse" and "fish" (point), anymore. Now, we are playing the "How-Big" game. This is the game with "small" and "big" (point). What game are we playing now? Right/No. We are playing the "How-Big" game now. This is the game with "small" and "big" (point)...") We explain how this feedback helps children and how training on the DCCS task with this feedback transfers to improve children's performance on the false belief task in the following section, "Modeling the Mutual Transfer between DCCS and False Belief Task (FB)".

In the FB training group, children were trained with two false statements (Hale & Tager-Flusberg, 2003) and one FB task (Wimmer & Perner, 1983) at each training session. After each trial, the experimenter provided positive and negative feedback about their answers. Therefore, children got feedback six times in total at both training days. Similar to the DCCS training group, the feedback emphasized that the question was about the protagonist's perspective, which was different from the children's own perspective (e.g., "...Where is the shell now? Who put it there? Was Ernie able to see this?... Right/No. Ernie did not see that. So, does Ernie really know

that the shell is in the red house? Right/No. Ernie does not know that the shell is in the red house now. Where does Ernie think the shell is? Right/No. Ernie still thinks that the shell is in the yellow tower...”).

In the control group, children were trained either with four relative clauses (Penner, 1999) or with three trials of a classic number conservation task (Piaget, 1965). Again, children got positive or negative feedback after each trial.

The results showed that there was a *transfer effect* from the DCCS task to the FB task, meaning that training children with the DCCS task by providing feedback significantly improved children’s performance on the FB task at the post-test session. Similarly, there was a *transfer effect* from the FB task to the DCCS task, meaning that training children with the FB task by providing feedback significantly improved children’s performance on the DCCS task at the post-test session. Moreover, there was a *training effect* of the DCCS task, meaning that training children on the DCCS task by providing feedback significantly improved children’s performance on the DCCS task at the post-test session. Importantly, these improvements were significantly greater than children’s improvement in the control group. Finally, although children’s performance on the false belief task improved in all conditions, there was only a significant improvement in the DCCS training group. Kloo and Perner argued that the insignificant improvement of the FB prediction score in the FB training group might be due to the fact that children’s scores were already good and there was little room for further improvements.

## **Modeling the mutual transfer between the DCCS task and the false belief task (FB)**

In this section, we first discuss the relevant mechanisms of the cognitive architecture PRIMs and explain our DCCS and false belief task models. Subsequently, we explain the underlying mechanism of *training* effect in both training groups and the underlying mechanism of the *transfer* effect from the DCCS task to the FB task and vice versa. After that, we present the results of our simulations by comparing them to

the experimental data from Kloo and Perner's (2003) training study. Finally, we introduce our models' predictions.

### **The relevant mechanisms of the cognitive architecture PRIMs**

The cognitive architecture PRIMs is built as a theory of skill acquisition and of transfer of skills. It adopts the mechanisms of the declarative memory of ACT-R, which is a hybrid symbolic/sub-symbolic production-based cognitive architecture (Anderson, 2007).

Similar to ACT-R, the factual knowledge is represented in the form of chunks in declarative memory (i.e., "The president of the USA is Barack Obama"). However, in addition to the chunks of factual information, PRIMs architecture has *operators* (represented as hexagons in Figure 4) and goals (represented as rounded rectangles in Figure 4) in declarative memory. Operators, like production rules in ACT-R, are in the form of IF-THEN rules (condition-action) and implement the instruction of the given task.

The PRIMs architecture breaks down the complex production rules of ACT-R, which represent procedural knowledge (i.e., how to drive a car), into a fixed number of smallest possible elements, named PRIMs. PRIMs only move, compare or copy information between modules (i.e., declarative, visual, motor modules) independent from the content of the information. For example, a *condition* PRIM checks if working memory is empty and an *action* PRIM copies the visual input to working memory independently from the content of the information. Operators combine these PRIMs together to perform a task. Figure 2 presents the global outline of the PRIMs architecture.

For instance, in Figure 3, the operators, which are represented by the colored nodes, represent the task-specific operators of the DCCS and FB models and combine the gray (condition) and white (action) nodes, which represent the task-general *condition-action* PRIMs. While the red colored nodes denote the operators of the DCCS model, the blue colored nodes denote the operators of the FB model. The yellow halos show the common PRIMs between the two tasks.

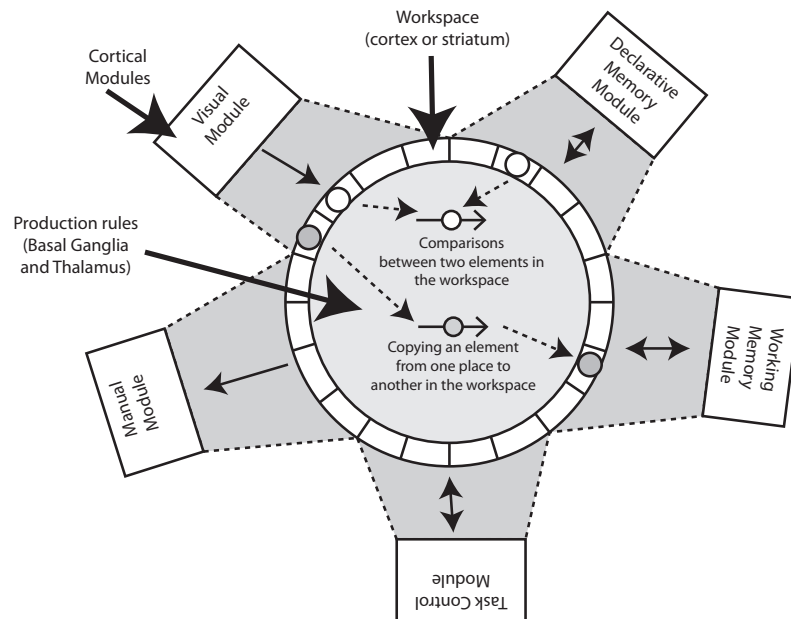


Figure 2. The global outline of the PRIMs architecture. Reprinted from “The Nature and Transfer of Cognitive Skills,” by N. A. Taatgen, 2013, *Psychological Review*, 120 (3), p. 443. Copyright 2013 by the American Psychological Association.

In the PRIMs architecture, a single task is implemented by multiple goals that can be reused for other tasks. Unlike ACT-R’s production rules, there is no hard connection between goals and operators in PRIMs (represented as dashed arrows in Figure 4), meaning that if a goal is triggered in a situation in which there are no associated operators, any matching operator can be tried. Current goals of the model activate operators to achieve those goals. If an operator is successful to complete a goal, the strength of association between the goal and the operator increases.

As we mentioned in the Introduction, the PRIMs architecture has two explanations that explain transfer of skills. Explanation 1 is based on the transfer of the task-general sequences of PRIMs. When a particular sequence of PRIMs is used often over time, it becomes more efficient to carry out that sequence. Whereas initially every PRIM is carried out individually, after learning the whole sequence of PRIMs is carried out in a single step (i.e., production compilation), considerably speeding up the process. Sequences of PRIMs are always task-general and can, therefore, be reused in other tasks. This means that if two tasks have common structural overlap, the PRIMs architecture can model knowledge transfer from one



task to another. However, Explanation 1 is based on a slow compilation process and therefore transfer occurs relatively slowly.

Explanation 2 is based on training a particular strategy, which is represented by operators. Operators, like other chunks in declarative memory, have base-level activations and associative strengths. After training a model with a task that forces the model to use a particular strategy (e.g., a proactive strategy), when the model is presented with another task that has two competing strategies (e.g., reactive vs. proactive), the model chooses the trained strategy (e.g., proactive) instead of the alternative competing strategy (e.g., a reactive strategy). Because Explanation 2 is based on the activations of the operators in declarative memory, transfer occurs faster than according to Explanation 1, which is based on utilities of PRIMs.

As can be seen from Figure 3, there is not so much overlap of *condition-action* PRIMs (gray and white nodes) between the DCCS and FB models and there is only one operator (i.e., *prepare*) that both models share. We argue that the key element of transfer from the DCCS task to the FB task and vice versa is based on Explanation 2 of the PRIMs architecture, which is training to choose a particular strategy, because the training time in the experiment is relatively short.

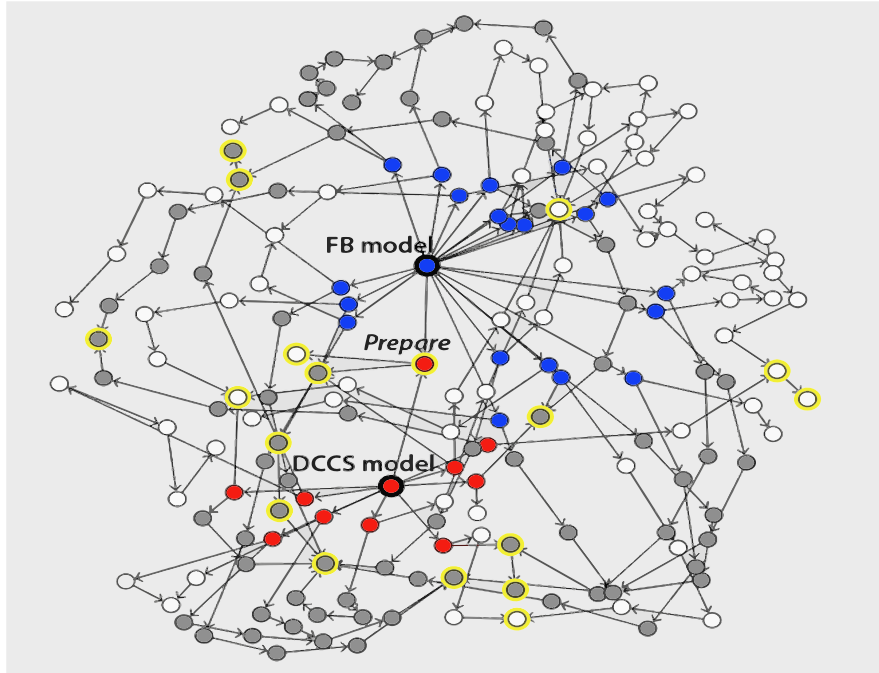


Figure 3. The representation of the operators (colored nodes) and *condition-action* PRIMs (gray and white nodes) in declarative memory for the FB model and the DCCS model. The yellow halos show the common PRIMs between the two tasks.

## A model of the Dimensional Change Card Sorting task (DCCS)

We constructed a model of the DCCS task without and with feedback representing Kloo and Perner's pre-test/post-test sessions and the training sessions, respectively. The DCCS model with feedback has an additional goal and an operator related to that goal that forces the model to prepare to use the *strategy of control* (see Figure 4a<sup>2</sup>). We explain how the model uses the *strategy of control* in detail below.

In line with Kloo and Perner's experiment, the DCCS model without feedback at pre-test and post-test sessions first plays six trials of the "Animal" game at the pre-switch phase and, after that, plays six trials of the "How-Big" game at the post-switch phase. Again, in line with Kloo and Perner's experiment, the DCCS model with feedback at the training sessions sorts the cards with five switches.

The steps that the DCCS model goes through over time as follows (cf., Buss & Spencer, 2008; Morton & Munakata, 2002; Marcovitch & Zelazo, 2000; van Bers, Visser, van Schijndel, Mandell, & Raijmakers, 2011):

1. The model starts with the goal "store-game" which spreads activation to the operators that put which game the model is playing into working memory (e.g., "Animal", "How-Big") and sends this to the declarative memory to be retrieved later if it is necessary (Figure 4a, O1 and O2).
2. Subsequently, there are two competing strategies to choose from after a test card is presented and before attending to a dimension of a presented test card. If the *default-attend* strategy is selected, meaning that the *default-attend* goal has a higher activation in the declarative memory, the model attends a dimension of the test card based on the pre-switch phase (e.g., the type of the animal) without checking what the game was (Figure 4a, O3). Therefore, while this strategy leads the model to a correct answer

---

<sup>2</sup> Note that the two models already have both a default strategy, which lacks of control, as well as a control strategy. This choice is based on the fact that Kloo and Perner's experimental results have shown that some children can pass these tasks even when they are presented to them for the first time. Our models are designed to represent an average child performing the tasks.

in the pre-switch phase, it does not work for the post-switch phase because the goal of the post-switch phase is to attend to the size of the animals instead of the animal type.

On the contrary, if the model selects the *prepare* strategy, it prepares itself to use the *strategy of control*, meaning that it changes the current goal to *control* (Figure 4a). Unlike the *default-attend* strategy, the *strategy of control* first requests a retrieval of the current game (i.e., “Animal” or “How-Big”), which was stored in the declarative memory at the beginning of the task (Figure 4a, O5). In this way, the model uses cognitive control by being flexible in behavior based on the current goal. After that, the DCCS model focuses its attention on a dimension based on the retrieved game (Figure 4a, O6 or O7). Therefore, when the model uses the *strategy of control*, it gives correct answers most of the time both at the pre-switch and post-switch phases. For example, if the game is a “How-Big” game at the post-switch phase and the test card is “small yellow horse”, the *default-attend* strategy focuses on “horse”, which is based on the pre-switch rule, “Animal” game. On the other hand, the *strategy of control* first retrieves what the game was (“How-Big”) and based on this retrieval, it focuses on the “How-Big” dimension of the test card, namely “small”.

3. After focusing on a dimension of the test card, the DCCS model makes a decision by requesting a retrieval of one of the decision chunks (i.e., “big yellow horse” on the left and “small red fish” on the right) from its declarative memory (Figure 4a, O8). For instance, at the post-switch phase, while the *default-attend* strategy (“attend-animal”) focuses its attention to the dimension “horse” and gives the wrong answer “left” after retrieving the decision chunk “horse left”, the *strategy of control* (“attend-howbig”) focuses its attention to the dimension “small” and gives the correct answer “right” after retrieving the decision chunk “small right”.

There is an additional mechanism of the DCCS model that leads the model to make errors when the retrieval of a decision is requested. It has been shown that there

is a visual clash between target and test cards (Doebel & Zelazo, 2015; Perner & Lang, 2002). For example, there is a visual clash between the picture of the big yellow horse on the target card and the small yellow horse on the test card when it needs to be sorted by size at the post-switch phase, which is after sorting the cards by animal type (pre-switch phase). In our DCCS model, this visual clash is represented by the strength of associations of chunks ( $S_{ji}$ ) in declarative memory. As a result of the visual clash, although the model selects the correct *prepare* strategy that prepares the model to use the *strategy of control*, it can still make errors during retrieval.

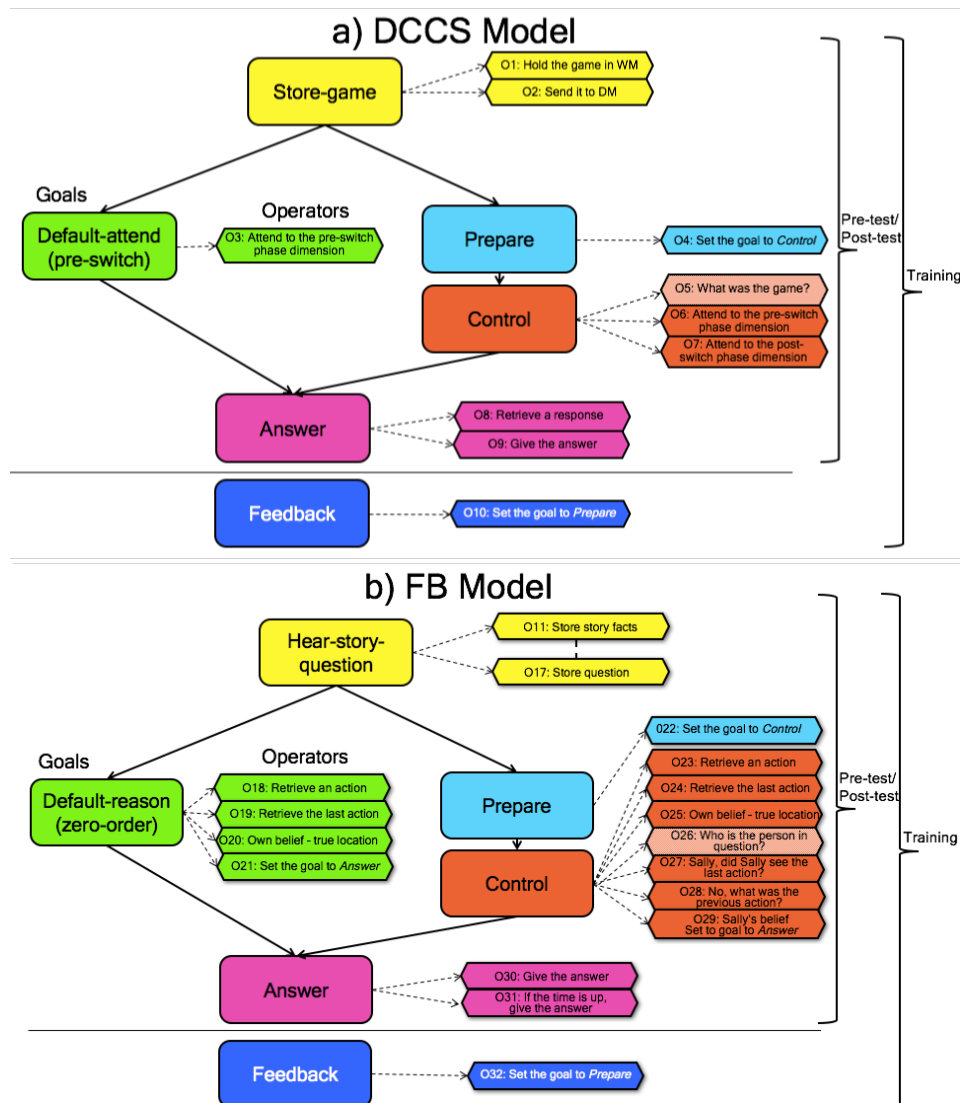


Figure 4. a) The DCCS model and b) the FB model at pre-test/post-test and training sessions. Note that bifurcations represent competing strategies<sup>3</sup> and the dashed arrows represent the operators related to the goals.

<sup>3</sup>Considering that children around the age of four start to use a strategy of control and that the tasks used in this study are novel tasks for children, we constructed our models with two different strategies (i.e., default and control). However, see Cohen, Servan-Schreiber, & McClelland (1992) for a framework proposing graded degrees of control.

Figure 5b shows an example of the associations between the “Animal” and “How-Big” types of chunks in the DCCS model. In addition to the positive associations with the same subgroup type of chunks (e.g., “horse – horse”), there are also positive associations between the subgroup of the “Animal” type of chunks and the subgroup of the “How-Big” type of chunks due to the visual clash (e.g., “horse – small”; “fish – big”). While the former positive associations lead the model to give correct answers when the correct strategy is selected, the latter positive associations represent the visual clash from the target cards and lead the model to make errors even if the correct strategy is selected. For example, at the post-switch phase, if the test card is “small yellow horse” and there is a target card on the left “big yellow horse” and on the right “small red fish”, even when the model uses the correct strategy (i.e., *strategy of control*) and focuses its attention on “small” according to the “How-Big” game, it can still give the wrong answer “left” instead of “right” based on the positive association between “small” and “horse”.



Figure 5. a) An example of the target and test cards in the DCCS task and b) an example of the associations between “Animal” and “How-Big” types of chunks in the DCCS model. The strengths of associations for the dimensions are set as follows: (horse horse 1.5), (fish fish 1.5), (small small 1.5), (big big 1.5), (big small -1.5), (horse fish -1.5), (horse big -1.0), (horse small 1.0), (fish small -1.0), (fish big 1.0).

### A model of the false belief task (FB)

Similar to the DCCS model, we constructed a model of the FB task without and with feedback representing Kloo and Perner’s pre-test/post-test sessions and the training sessions, respectively. The FB model with feedback has an additional operator, which is associated to the goal “Feedback”. This additional operator forces the model to prepare to use the *strategy of control* whenever the model presented with feedback, meaning that it changes the current goal to *control* goal (see Figure 4b).

The steps that the FB model goes through over time as follows (cf., Bello & Cassimatis, 2006; Goodman et al., 2006; Hiatt & Trafton, 2010):

1. First, the story facts that include actions (e.g., “Ernie put the shell in the yellow tower”, “The bear put the shell in the red house”) and the false belief question (i.e., “Where does Ernie think the shell is?”) are presented on the screen one by one. The operators that are associated with the “Hear-story-questions” (Figure 4b, O11 – O17) put those facts into working memory and send them to declarative memory by chaining them together to be retrieved later when necessary. After being presented with the FB question, the model starts reasoning.
2. Similar to the DCCS model, the FB model has two competing strategies to choose from before starting to reason about the presented false belief question (i.e., “Where does Ernie think the shell is?”). The *default-reason* strategy gives an answer based on the model’s own perspective (reality/zero-order reasoning strategy) without checking who is the question in person (i.e., as if the question was “Where is the shell” instead of the false belief question “Where does Ernie think the shell is?”). This strategy requests a retrieval of an action<sup>4</sup> (Figure 4b, O18). If the retrieved action is not the last action, an operator requests the retrieval of a last action (Figure 4b, O19). When the last action is retrieved (“The bear put the shell in the red house”), the model creates a “belief” chunk in working memory (Figure 4b, O20) about the location of the object (“in the red house”) and gives an answer based on its own belief (Figure 4b, O30).

Alternatively, if the model selects the *prepare* strategy, it prepares the model using the *strategy of control* by changing its goal to “Control” (Figure 4b, O22). The *strategy of control* starts with reasoning from the model’s own perspective as in the *default-reason* strategy (Figure 4b, O23 – O25). However, subsequently, it requests a retrieval about the person in

---

<sup>4</sup> We used the action of moving the shell, but the model could also easily be adapted for seeing.

question (“Ernie”) instead of giving an answer based on its own perspective (Figure 4b, O26). Note that this procedure is very similar to the DCCS models’ *strategy of control*, which first checks what the game was instead of the *default-attend* strategy that does not have an element of control (Figure 4a, O5).

3. After retrieving that the question is about “Ernie”, the FB model requests a retrieval whether “Ernie” saw the shell in the location that is in its working memory (“in the red house”, Figure 4b, O27). This retrieval request leads to a retrieval error. Based on this retrieval error, the model “infers” that “Ernie does not know that the shell is in the red house” and requests a retrieval of a chunk that includes “Ernie” and an action (Figure 4b, O28). Finally, the model retrieves the chunk “Ernie put the shell in the yellow tower” and creates a “belief” chunk in its working memory that “Ernie believes that the shell is in the yellow tower” (Figure 4b, O29) and gives the correct answer “in the yellow tower” (Figure 4b, O30).

In addition to selecting the wrong *default-reason* strategy, the FB model has another mechanism that leads the model to make errors. This mechanism is due to the time threshold of the FB model (i.e., 28 seconds). If the model’s run-time passes the preset threshold, the model stops reasoning and gives the location that it currently has in working memory as an answer (Figure 4b, O31). In this way, we simulate that the model gives up reasoning for any reason (e.g., it takes too long or it gets distracted). The idea of a time threshold when children are performing a task is consistent with research showing that children perform better in language comprehension tasks and cognitive tasks when they are given more time (Ling, Wong, & Diamond, 2015; van Rij, van Rijn, & Hendriks, 2010; Hendriks, van Rijn, & Valkanier, 2006; Diamond, Kirkham, & Amso, 2002). When the model uses the *strategy of control*, first it starts to reason from its own perspective and puts into working memory the location where the shell really is (reality). If the time threshold is reached and if the model has not reasoned about Ernie’s perspective yet, it gives the answer “in the red house” instead of the answer “in the yellow tower”.

## The underlying mechanism of the training effects

As we mentioned before, we use the term *training effect* to refer both to the improvement of the DCCS task after training on the DCCS task with feedback and to the improvement of the FB task after training on the FB task with feedback.

Similar to Kloo and Perner's experiment, both the DCCS and FB models receive feedback after each trial at the training sessions. Note that in the DCCS training group, the feedback forces children to first check which game they are playing before making a decision (i.e., "What game are we playing now?"). In the FB training group, the feedback urges children to take the perspective of the protagonist who is in question (e.g., "...Does Ernie really know that the shell is in the red house? Where does Ernie think the shell is? ..."). Therefore, the feedback forces children to use a *strategy of control* in the both DCCS and FB training groups.

The feedback in training sessions is represented as follows for both the DCCS and FB models: i) The screen that the model "sees" presents the word "feedback"; ii) the operator "feedback" (Figure 4a, O10; Figure 4b, O32) which is associated to the goal "Feedback" matches the current state of the model and puts the goal "Prepare" into one of the goal slots. With repetition, this procedure increases the activation of the *prepare* strategy, which forces the model to use the *strategy of control*. Therefore, the DCCS model starts to use a *strategy of control* by first checking what the game is instead of the *default-attend* strategy and the FB model starts to use the *strategy of control* by taking the perspective of the person in question instead of giving an answer based on its own perspective.

In this way, in the DCCS training group, after training the DCCS model with feedback, the accuracy of the DCCS model at post-test becomes higher than the DCCS model at pre-test. Similarly, in the FB training group, after training the FB model with feedback, the accuracy of the FB model at post-test becomes higher than the FB model at pre-test.



## **The underlying mechanism of mutual transfer between the DCCS and FB tasks**

In the previous three subsections, we explained how the FB and DCCS models work and we delineated the underlying mechanisms of the training effect. In this subsection, we explain the underlying mechanism by which our models show transfer from the DCCS task to the FB task and vice versa.

As we mentioned before, there is not so much overlap in the procedural knowledge between the FB and DCCS models (Figure 3, the yellow halos). The key element of the mutual transfer between the DCCS and the FB models is based on the PRIMs architecture's Explanation 2, which is training a particular strategy.

As shown in Figure 4a and Figure 4b, the FB and DCCS models have a common structure. There are two competing strategies, only one of which leads the model to give a correct answer. While the *default-attend* and *default-reason* strategies lack the element of control, the *prepare* strategy, which is trained by the explicit feedback, forces the model use the *strategy of control* (see Taatgen, 2013 for a similar modeling approach to the transfer between Stroop task and task switching). Thus, once the DCCS model has been trained on the FB model with feedback, the activation of the *prepare* strategy gets higher. Because the *prepare* strategy's activation increases, at the post-test session, the DCCS model selects the *prepare* strategy instead of the competing *default-attend* strategy after training on the FB model with feedback. Therefore, the DCCS model's accuracy gets higher at the post-test session when it has been trained on the FB model with feedback. Similarly, once the FB model has been trained on the DCCS model with feedback, again the activation of the *prepare* strategy gets higher. Therefore, the FB model selects the *prepare* strategy instead of the competing *default-reason* strategy and the FB model's accuracy gets higher at the post-test session.

## **Results of the DCCS and FB models and comparison to experimental data**

Similar to Kloo and Perner's study, we ran simulations in three training groups (DCCS training, FB training, control). Table 1 shows the protocol to obtain the results

of the DCCS and FB models for each training group at pre-test, training and post-test sessions. We repeated the protocol in Table 1 for 15 times at each training group. In this way, we aimed the results to represent 15 children performing the tasks at each training group. For example, the accuracies of the DCCS and FB models at each pre-test and post-test sessions are based on a total of 1,500 repetitions ( $15 * 100$ ) in each training group.

As can be seen from Figure 6a and Figure 6b, our results have similar patterns with Kloo and Perner's results in terms of training effect and transfer effect. Importantly, similar to Kloo and Perner's findings, the improvements of both the DCCS and FB tasks in the DCCS training group is higher than their improvements in the FB training group. Based on our results, we predict that the reason of this difference is due to the unequal number of times of feedback that was provided in the DCCS and FB training groups. Note that Kloo and Perner's experiment trained children ten times with feedback in the DCCS training group. On the other hand, children were trained six times with feedback in the FB training group. We simulated this by running 40 repetitions at the training session of the FB training group (40 times feedback) and 30 repetitions at the training session of the DCCS training group ( $30 * 5$  times feedback). The higher number of repetitions in the DCCS training group makes the selection of the *strategy of control* more likely than the *default* strategies due to increased activation. Note that the only parameter that is fitted to the experimental results is the number of repetitions of the FB training group. If the number of repetitions of the FB training group is set proportional to Kloo and Perner's experimental design (i.e., 90), the improvement of the models in this training group will be higher than children's improvement in the experiment.